What is claimed is:

1. A method to reduce latency in accessing a memory from a bus, the method comprising:

pre-fetching a plurality of data from the memory to a cache queue in response to a request; and

delivering the pre-fetched data from the cache queue to the bus independently of the memory.

2. The method of claim 1 wherein pre-fetching comprises:

determining if an amount of data in the cache queue is above a predetermined level; and

placing the request to a memory controller controlling the memory if the amount of data is not above the predetermined level, the request causing the memory controller to transfer the plurality of data to the cache queue, the request being buffered in a request queue.

3. The method of claim 2 wherein delivering comprises:

transferring the data from the cache queue to the bus if the data in the cache queue is ready.

4. The method of claim 1 further comprising:

determining if the request is valid; and

processing a cache miss request if the request results in a cache miss.

5. The method of claim 4 wherein processing the cache miss request comprises:

providing a purge signal;

marking an entry in a scheduler according to the purge signal;

purging data corresponding to the marked entry; and

placing the request to the memory controller.

6. The method of claim 5 wherein the bus is a peripheral component interconnect (PCI) bus.

7. The method of claim 6 wherein the request is one of a 32-byte and a 64-byte requests.

8. An apparatus to reduce latency in accessing a memory from a bus, the apparatus comprising:

a pre-fetcher to pre-fetch a plurality of data from the memory to a cache queue in response to a request; and

a cache controller coupled to the cache queue and the pre-fetcher to deliver the pre-fetched data from the cache queue to the bus independently of the memory.

9. The apparatus of claim 8 wherein the pre-fetcher comprises:

a watermark monitor to determine if an amount of data in the cache queue is above a predetermined level;

a request packet generator coupled to the watermark monitor to place the request to a memory controller controlling the memory if the amount of data is not above the predetermined level, the request causing the memory controller to transfer the plurality of data to the cache queue; and

a request queue coupled to the request packet generator to store the request provided by the request packet generator.

10. The apparatus of claim 9 wherein the cache controller transfers the data from the cache queue to the bus if the data in the cache queue is ready.

11. The apparatus of claim 8 further comprising:

a peripheral bus controller coupled to the bus and the pre-fetcher to determine if the request is valid;

a data coherence controller coupled to the pre-fetcher to provide a purge signal when the request corresponds to a cache miss; and

a scheduler coupled to the request queue and the data coherence controller to store entries corresponding to the requests, the entries being marked according to the purge signal from the data coherence controller.

12. The apparatus of claim 11 further comprising:

a data mover coupled to the cache queue and the scheduler to transfer data from the memory to the cache queue, the data mover purging data corresponding to a marked entry from the scheduler.

13. The apparatus of claim 12 wherein the bus is a peripheral component interconnect (PCI) bus.

14. The apparatus of claim 13 wherein the request is one of a 32-byte and a 64-byte requests.

15. A system comprising:

a memory;

a bus; and

a bus access circuit coupled to the memory and the bus to reduce latency in accessing the memory from the bus, the circuit comprising:

a pre-fetcher to pre-fetch a plurality of data from the memory to a cache queue in response to a request, and

a cache controller coupled to the cache queue and the pre-fetcher to deliver the pre-fetched data from the cache queue to the bus independently of the memory.

16. The system of claim 15 wherein the pre-fetcher comprises:

a watermark monitor to determine if an amount of data in the cache queue is above a predetermined level;

a request packet generator coupled to the watermark monitor to place the request to a memory controller controlling the memory if the amount of data is

not above the predetermined level, the request causing the memory controller to transfer the plurality of data to the cache queue; and

a request queue coupled to the request packet generator to store the request provided by the request packet generator.

17. The system of claim 16 wherein the cache controller transfers the data from the cache queue to the bus if the data in the cache queue is ready.

18. The system of claim 15 wherein the bus access circuit further comprises:

a peripheral bus controller coupled to the bus and the pre-fetcher to determine if the request is valid;

a data coherence controller coupled to the pre-fetcher to provide a purge signal when the request corresponds to a cache miss; and

a scheduler coupled to the request queue and the data coherence controller to store entries corresponding to the requests, the entries being marked according to the purge signal from the data coherence controller.

19. The system of claim 18 wherein the bus access circuit further comprising:

a data mover coupled to the cache queue and the scheduler to transfer data from the memory to the cache queue, the data mover purging data corresponding to a marked entry from the scheduler.

20. The system of claim 19 wherein the bus is a peripheral component interconnect (PCI) bus.

21. The system of claim 20 wherein the request is one of a 32-byte and a 64-byte requests.